IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)
	)
Adrian E. COLLEY et al.	) Group Art Unit: 2194
	)
Application No.: 09/753,686	) Examiner: Van H. Nguyen
	)
Filed: January 4, 2001	)
	)
For: METHOD AND SYSTEM FOR	) Confirmation No.: 1658
PASSING OBJECTS IN A	)
DISTRIBUTED SYSTEM USING	)
SERIALIZATION CONTEXTS	)

Attention: **Mail Stop Appeal Brief-Patents**
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

AMENDED APPEAL BRIEF UNDER BOARD RULE § 41.37(d)

In support of the Notice of Appeal filed June 15, 2006, Appellants filed an Appeal

Brief under 37 C.F.R. § 41.37 on August 23, 2006. In response, the Examiner issued a

Notice of Non-Compliant Appeal Brief on November 13, 2006 alleging that the

"Summary of Claimed Subject Matter" section required by 37 C.F.R. § 41.37(c)(v) was

deficient. Although Appellants maintain that the Appeal Brief filed on August 23, 2006

fully complies with the applicable statutes and rules, Appellants file this Amended

Appeal Brief in order to maintain the appeal pursuant to 37 C.F.R. § 41.37(d). In the

Amended Appeal Brief, only the "Summary of Claimed Subject Matter" section differs

from the as-filed Appeal Brief.

If any fee is required, Appellants request that it be charged to Deposit Account

No. 06-0916.

Appellants maintain their appeal of the final rejections of claims 1-9, 15, 17, and

19-22 set forth in the Final Office Action mailed February 16, 2006.

TABLE OF CONTENTS

Real Party In Interest

SUN MICROSYSTEMS, Inc. is the real party in interest, as evidenced by an

Assignment recorded at reel 011420, frame 0567, on January 4, 2001.

Related Appeals and Interferences

There are currently no other appeals or interferences, of which Appellants,

Appellants' legal representative, or Assignee are aware, that will directly affect or be

directly affected by or have a bearing on the Board's decision in the pending appeal.

Status Of Claims

Claims 1-9, 15, 17, and 19-22 have been finally rejected and are under appeal.

Claims 10-14, 16, and 18 have been cancelled. Pursuant to 37 C.F.R.

§ 41.37(c)(1)(viii), a listing of the claims under appeal is included in the attached Claims

Appendix.

Status Of Amendments

An Amendment After Final was filed on May 15, 2006 and was entered,

according to the Advisory Action mailed on May 24, 2006.

Summary Of Claimed Subject Matter

Claim 1 recites a method in a distributed system for passing a first object and a second object, wherein the first object and the second object are instances of a class. (See, e.g., Specification, p. 5, ll. 1-3.) The first object is passed from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor. (See, e.g., Specification, p. 8, ll. 2-11; Fig. 5, step 510.) The recipient stores the handle and the descriptor received from the sender with the first object. (See, e.g., Specification, p. 9, l. 21 - p. 10, l. 8; Fig. 5, step 512.)

The sender may then pass the second object to the recipient with the handle. (See, e.g., Specification, p. 9, ll. 2-5; Fig. 5, step 506 ("**Sender Sends to Recipient Handle, Data**")) As explained in the specification, "[t]he first object is passed from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor. The handle and the descriptor are stored by the recipient. **The second object is then passed from the sender to the recipient with the handle**, and the recipient uses the handle to determine the descriptor." (Specification, p. 5, ll. 1-7.) Still further, in another embodiment the specification provides: "**the sender...sends the handle rather than the full class descriptor to the recipient (step 306).**" (Specification, p. 11, ll. 1-4.)

Finally, in the method of claim 1, the recipient uses the handle received with the second object to access the descriptor received by the recipient with the first object. (See, e.g., Specification, p. 11, ll. 4-6 ("**[t]he recipient then uses the handle to look up the class descriptor**"); Fig. 3, step 308.) As another example of support for this claim element, the specification provides: "[t]he first object is passed from a sender to a

recipient with a descriptor of the class and a handle corresponding to the descriptor. The handle and the descriptor are stored by the recipient. The second object is then passed from the sender to the recipient with the handle, and **the recipient uses the handle to determine the descriptor**." (Specification, p. 5, ll. 1-7.) Still further, the specification states, "**[t]he recipient then uses the handle to look up the class descriptor....**" (Specification, p. 11, ll. 4-5.)

Claim 5 recites a method in a distributed system for passing a first object and a second object to a recipient, wherein the first object and the second object are instances of a class. (See, e.g., Specification, p. 5, ll. 1-3.) A sender passes the first object to the recipient with a descriptor of the class and a handle corresponding to the descriptor. (See, e.g., Specification, p. 8, ll. 2-11; Fig. 5, step 510.)

Then, the sender passes the second object to the recipient with the handle, whereupon receipt by the recipient, the recipient uses the handle received with the second object to access the descriptor of the class received with the first object. (See, e.g., Specification, p. 9, ll. 2-5; p. 11, ll. 4-6 ("**[t]he recipient then uses the handle to look up the class descriptor**"; Fig. 5, step 506 ("**Sender Sends to Recipient Handle, Data**"); Fig. 3, step 308.) As explained in the specification, "[t]he first object is passed from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor. The handle and the descriptor are stored by the recipient. **The second object is then passed from the sender to the recipient with the handle, and the recipient uses the handle to determine the descriptor**." (Specification, p. 5, ll. 1-7.) Furthermore, the specification explains that in certain embodiments, "**the**

9

sender...sends the handle rather than the full class descriptor to the recipient

(step 306)." (Specification, p. 11, ll. 1-4.)

Claim 7 recites a method in a distributed system for interpreting a first object and

a second object, wherein the first object and the second object are instances of a class.

(See, e.g., Specification, p. 5, ll. 1-3.) The first object is received from a sender with a

descriptor of the class and a handle corresponding to the descriptor. (See. e.g.,

Specification, p. 13, l. 4-7; Fig. 5, step 510.) The handle and the descriptor are stored.

(See, e.g., Specification p. 13, ll. 7-8; Fig. 5, step 512.) The second object is received

with the handle, and the handle received with the second object is used to access the

descriptor received with the first object. (See, e.g., Specification, p. 13, ll. 21-22; Fig. 3,

step 308.) As explained in the specification, "[t]he first object is passed from a sender to

a recipient with a descriptor of the class and a handle corresponding to the descriptor.

The handle and the descriptor are stored by the recipient. **The second object is then**

passed from the sender to the recipient with the handle, and the recipient uses

the handle to determine the descriptor." (Specification, p. 5, ll. 1-7.) The

specification further provides: "**the sender...sends the handle rather than the full**

class descriptor to the recipient (step 306). The recipient then uses the handle to

look up the class descriptor...." (Specification, p. 11, ll. 1-5.)

Claim 15 recites a distributed system comprising a client computer and a server

computer. (See, e.g. Specification, Fig. 1.) The client computer of claim 15 comprises

a memory with a client program (e.g., Fig. 1, elements 108, 126) that sends an object of

a class to a remote location together with a handle corresponding to a descriptor of the

class, and with an outgoing serialization context that stores the descriptor of the class

and the handle corresponding to the descriptor. (See, e.g., Specification p. 9, ll. 7-19; Fig. 1, element 126; Fig. 2, element 202.) The client computer of claim 15 also comprises a processor that runs the client program. (See, e.g., Specification, Fig. 1, element 112.) The server computer of claim 15 comprises a memory (e.g., Fig. 1, element 128) with an incoming serialization context that stores the descriptor of the class and the handle received from the client computer before the object was sent, and with a server program (e.g., Fig. 1, element 148) that receives the object from the client program and that uses the handle received with the object to access the descriptor of the class in the incoming serialization context. (See, e.g., Specification, p. 9, ll. 7-21; Fig. 1, element 146; Fig. 2, element 208.) As explained in the specification, "**the sender...sends the handle rather than the full class descriptor to the recipient (step 306). The recipient then uses the handle to look up the class descriptor....**" (Specification, p. 11, ll. 1-5.) The client computer of claim 15 also comprises a processor that runs the server program. (See, e.g., Specification, Fig. 1, element 132.)

Claim 17 recites a computer-readable medium containing instructions (e.g., Fig. 1, element 126 or 146) for controlling a data processing system to perform a method, the method for sending a first object and a second object from a source to a destination, wherein the first object and the second object are instances of a class. (See, e.g., Specification, p. 6, ll. 5-10; p. 9, ll. 8-17.) The method comprises sending the first object from the source to the destination with a descriptor of the class and a handle corresponding to the descriptor. (See, e.g., Specification, p. 8, ll. 2-11; Fig. 5, step 510.) The handle and the descriptor received from the source by the destination are stored. (See, e.g., Specification, p. 9, l. 21 - p. 10, l. 8; Fig. 5, step 512.)

The second object is sent from the source to the destination with the handle. (See, e.g., Specification, p. 9, ll. 2-5; Fig. 5, step 506 ("**Sender Sends to Recipient Handle, Data**")) As the specification explains, "[t]he first object is passed from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor. The handle and the descriptor are stored by the recipient. **The second object is then passed from the sender to the recipient with the handle**, and the recipient uses the handle to determine the descriptor." (Specification, p. 5, ll. 1-7.) The specification also provides: "**the sender...sends the handle rather than the full class descriptor to the recipient** (step 306)." (Specification, p. 11, ll. 1-4.)

Finally, the handle received by the destination with the second object is used to access the descriptor received by the destination with the first object. (See, e.g., Specification, p. 11, ll. 4-6 ("**[t]he recipient then uses the handle to look up the class descriptor**"); Fig. 3, step 308.) The specification provides: "[t]he first object is passed from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor. The handle and the descriptor are stored by the recipient. The second object is then passed from the sender to the recipient with the handle, and **the recipient uses the handle to determine the descriptor**." (Specification, p. 5, ll. 1-7.) The specification also states, in certain embodiments, that "**[t]he recipient then uses the handle to look up the class descriptor....**" (Specification, p. 11, ll. 4-5.)

Grounds of Rejection

Claims 1-9, 15, 17, and 19-22 stand rejected under 35 U.S.C. § 102(b) as being anticipated by P*elegri-Llopart et al.*, U.S. Patent No. 5,999,988.

The Examiner also rejected claims 19 and 21 in the Final Office Action under 35 U.S.C. § 112, 2nd paragraph, as being indefinite for lacking proper antecedent basis. In the Amendment After Final, Appellants proposed amending claims 19 and 21 to correct clerical errors in the claims. In the Advisory Actions dated May 24, 2006 and June 19, 2006, the Examiner indicated that the amendments were entered and did not further mention the section 112 rejections. Therefore, Appellants understand that claims 19 and 21 are no longer rejected under 35 U.S.C. § 112.
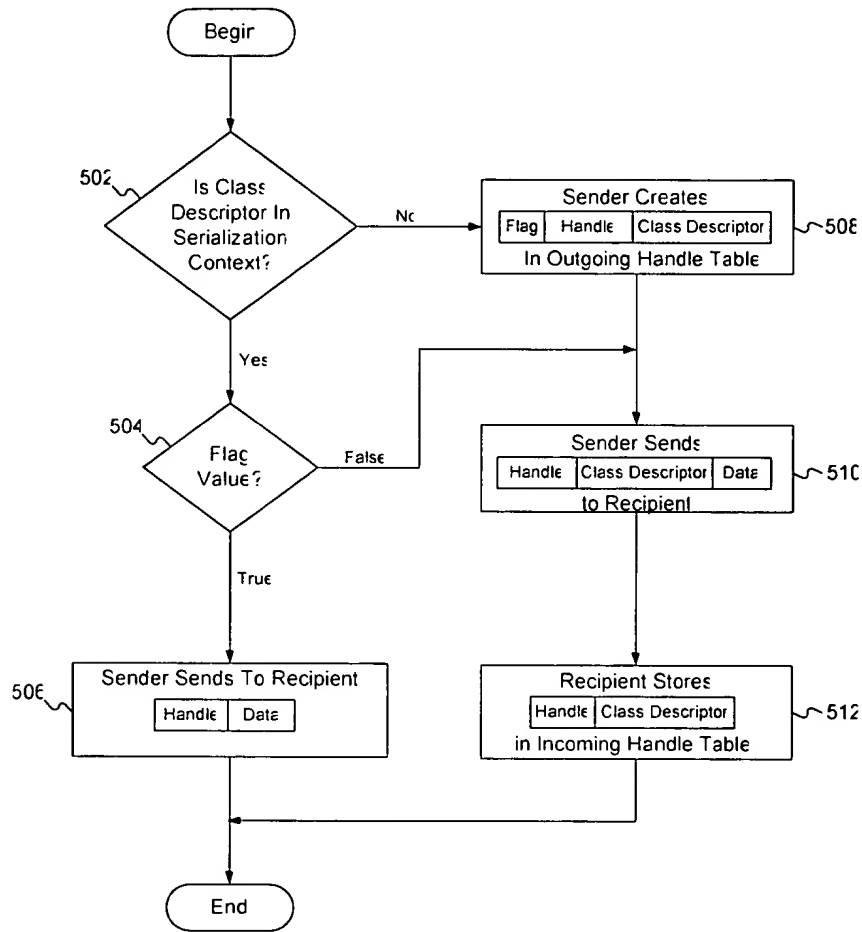
<u>Argument</u>

Claims 1-9, 15, 17, and 19-22 stand rejected under 35 U.S.C. § 102(b) as being

anticipated by U.S. Patent No. 5,999,988 to *Pelegri-Llopart et al.* To anticipate a claim,

the reference must teach every element of the claim. M.P.E.P. § 2131.01 (8th ed. 2001,

revised August 2005). Because the Examiner has not shown that the reference teaches

every element recited in the claims, the rejections are improper and should be reversed.

I.   Overview of claimed subject matter.

The present claims recite systems and methods for passing and receiving

objects in a distributed system. Serialized versions of objects, such as parameters or

return values, are passed in byte streams across a network. (See., e.g., Specification,

p. 7, l. 10 - p. 8, l. 2.) A serialized object contains object data and a class descriptor,

which describes the content and the format of the object data. When a serialized object

is passed, the object data and the class descriptor are transmitted across a network.

(See, e.g., Specification, p. 4, ll. 3-7; p. 8, ll. 1-11.)

To reduce the number of redundant class descriptors sent with objects, a

"serialization context" can be used to pass the class descriptors with serialized objects.

A serialization context is a dictionary object mapping a class descriptor to a

corresponding integer handle. (See, e.g., Specification, p. 9, ll. 2-6.) Using serialization

contexts, after a full class descriptor is sent with the first object of a type, just the integer

handle may be sent with subsequent objects of that type, saving processing time. (See,

e.g., Specification, p. 9, ll. 2-6.)

An exemplary process for passing objects using serialization contexts is shown in

Figure 5 of the application, reproduced below:

In the process of Figure 5, the sender first checks to see if the class descriptor is

already stored in a serialization context (step 502). If so, then the sender checks the

value of a corresponding committed flag (step 504). If the committed flag is true, the

sender can send just the object data and the handle, knowing that the class

descriptor/handle pair is already stored in the recipient's serialization context (step 506).

If the class descriptor is not stored in the serialization context, then the sender creates a

new serialization context entry, with a new handle and a committed flag set to false, and

sends the object data with the new handle and class descriptor to the recipient (step

510).

II.   Teachings of the *Pelegri-Llopart et al.* patent.

In the system of *Pelegri-Llopart et al.*, a client application on a first virtual

machine communicates with a remote object on a second virtual machine by generating

a local stub object that implements only those interfaces supported by the first virtual

machine. (*Pelegri-Llopart et al.*, col. 4, ll. 27-32; col. 5, ll. 64-66; Fig. 1.) Patent figure

10, reproduced below, depicts the run-time stub generation disclosed in the reference.

RECEIVE INTERFACE DESCRIPTOR (LIST OF INTERFACES)
OF REMOTE OBJECT   803

RECEIVE OBJECT HANDLE OF REMOTE OBJECT   804

COMPARE INTERFACES OF REMOTE OBJECT IDENTIFIED BY
THE INTERFACE DESCRIPTOR (LIST OF INTERFACES) WITH
INTERFACES SUPPORTED BY THE FIRST VIRTUAL
MACHINE   808

GENERATE A COMMON INTERFACE LIST THAT INCLUDES
INTERFACES THAT ARE IDENTIFIED BY THE INTERFACE
DESCRIPTOR AND ARE SUPPORTED BY THE FIRST
VIRTUAL MACHINE
810

GENERATE A METHOD LIST THAT INCLUDES THE METHODS
SUPPORTED BY THE COMMON INTERFACES
812

GENERATE AT RUN-TIME A STUB CLASS THAT
REPRESENTS IN THE FIRST VIRTUAL MACHINE THE
REMOTE OBJECT WHICH IS IMPLEMENTED IN A SECOND
VIRTUAL MACHINE BASED ON THE COMMON INTERFACE
LIST AND THE METHOD LIST   816

INSTANTIATE STUB CLASS WITH INTERFACE DESCRIPTOR
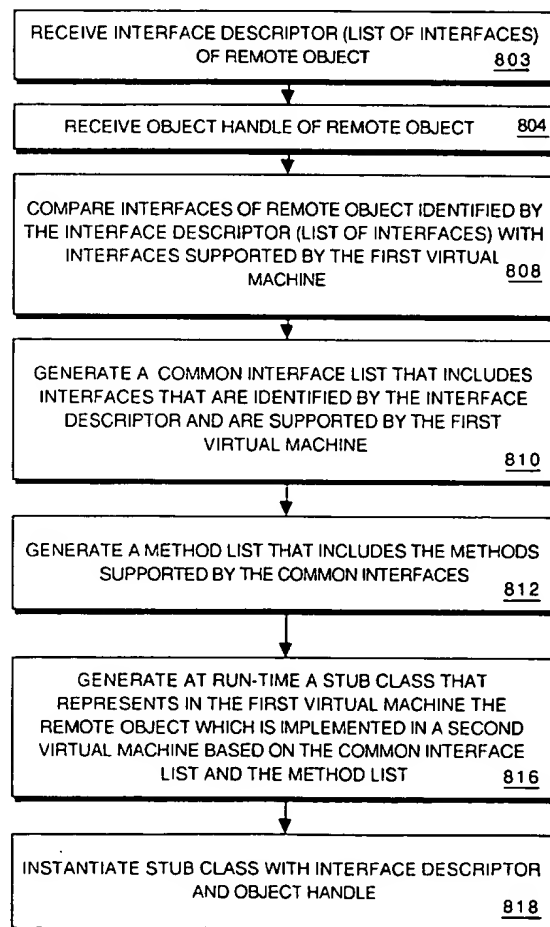AND OBJECT HANDLE
818

FIG. 10

As the patent explains, a runtime stub generator on the first virtual machine

receives an object handle (e.g., an address of a remote object) and an interface

descriptor (i.e., a list of interfaces implemented by the remote object). (*Id.*, Fig. 10,

steps 803 and 804; col. 10, II. 19-24.) The runtime stub generator then compares the

interface descriptor to a list of interfaces supported by the first virtual machine, and

generates a stub class that "represents in the local machine the remote object that is

implemented in the second virtual machine...." (*Id.*, Fig. 10, steps 808, 810, and 816;

col. 4, II. 25-29; col. 10, II. 34-40.) The local stub class instance can be used by a

process on the first virtual machine to invoke member functions of the remote object.

(*Id.*, col. 4, II. 45-55.)

III. Claims 1-6, 15, 17, and 19-22 are not anticipated by *Pelegri-Llopart et al.* because the reference does not disclose passing a first object from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor.

Claim 1 recites a method in a distributed system for passing a first object and a

second object, wherein the first object and the second object are instances of a class,

including, among other things, passing the first object from a sender to a recipient with a

descriptor of the class and a handle corresponding to the descriptor. In this step, three

things are passed from the sender to the recipient: (1) the first object that is an instance

of a class; (2) a descriptor of the class; and (3) a handle corresponding to the

descriptor. (See, e.g., Specification, p. 5, II. 3-4; Fig. 5, step 510.)

The Examiner has not shown that *Pelegri-Llopart et al.* discloses the claimed

method, including at least this claim element. Instead, as the Examiner noted in the

Final Office Action, *Pelegri-Llopart et al.* discloses passing only an "object reference

[that] includes an interface descriptor and an object handle associated with the

object...." (Final Office Action, p. 3; *Pelegri-Llopart et al.*, col. 8, II. 57-60.) Rather than

being passed, the remote object remains "implemented in a second virtual machine...."

(*Pelegri-Llopart et al.*, Fig. 1; col. 8, II. 1-8; col. 6, II. 54-55.)

In the Advisory Action mailed on June 19, 2006, the Examiner asserted that "the

limitation 'passing the first object' and 'passing the second object' as claimed and

according to the specification does not actually pass the object, but only passes data

representing the object and recreating the object at the recipient using the data

representing the object." (6/19/06 Advisory Action, p. 2.) However, the specification

clearly describes embodiments of the invention in which "objects can be passed

between client computer 102 and server computer 104." (Specification, p. 7, II. 10-11,

emphasis added.) One exemplary method passes objects using byte streams

containing "serialized versions of Java™ objects, e.g. parameters or return values."

(Specification, p. 7, I. 11 - p. 8, I. 2.) For example, "[w]ithin a single remote method call,

a class descriptor is sent with the first object of that type that is serialized...." (*Id.*, p. 8,

II. 8-9., (emphasis added); Fig. 5, step 510.)

The Examiner further argued that "Pelegri-Llopart teaches sending "data

representing the object" by interpreting the interface descriptor of the reference as data

representing the object. (6/19/06 Advisory Action, p. 2, ¶ 3.) However, the Examiner

also argued that "Pelegri-Llopart discloses...class descriptor [interface descriptor, i.e.,

col. 7, line 55 - col. 8, line 67]...." (*Id.*) Thus, the Examiner attempts to show that the

interface descriptor of *Pelegri-Llopart et al.* teaches both the claimed first object, i.e., an

instance of a class, and the claimed descriptor of that class. This unreasonable claim

interpretation should not be allowed to stand because it would be inconsistent with the plain language of the claims, which recite passing the first object (i.e., an instance of a class) with a descriptor of the class and a handle corresponding to the descriptor. "During patent examination, the pending claims must be 'given their broadest reasonable interpretation consistent with the specification.'" M.P.E.P. § 2111 (8th ed. 2001, revised August 2005, emphasis added, internal citations omitted.)

As clearly explained in the *Pelegri-Llopart et al.* reference, what is passed in the reference is "an interface descriptor and an object handle associated with the object...." The remote object itself is not passed. (*Pelegri-Llopart et al.*, col. 8, ll. 57-60.) Thus, the Examiner has not shown a teaching of passing the first object from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor, as recited in claim 1.

Because the Examiner has not shown a teaching of every element of claim 1 in *Pelegri-Llopart et al.*, the section 102 rejection of claim 1 is improper and Appellants request its reversal. Claims 2-4 and 19-20 depend from claim 1 and therefore incorporate its recitations. Because the Examiner has not shown a teaching in the reference of every element of claim 1, he also has not shown a teaching of every element of dependent claims 2-4 and 19-20, and Appellants request the reversal of the section 102 rejections of claims 2-4 and 19-20.

Claims 5, 15 and 17, although varying in scope, contain similar recitations to those discussed above with respect to claim 1. Therefore, at least for the reasons discussed above with reference to claim 1, the Examiner has not shown a teaching of these elements of claims 5, 15 and 17 in *Pelegri-Llopart et al.* Therefore, the section

102 rejections of claims 5, 15 and 17, and their respective dependent claims, should

also be reversed.

IV.   Claims 1-6, 15, 17, and 19-22 are not anticipated by Pelegri-Llopart because the reference does not disclose passing a second object from the sender to the recipient with the handle.

The method of claim 1 recites a combination of steps, including a step of passing

the second object from the sender to the recipient with the handle. The Examiner has

not shown that *Pelegri-Llopart et al.* teaches the claimed subject matter, including this

step. Instead, the Examiner again noted that *Pelegri-Llopart et al.* discloses passing an

"object reference [that] includes an interface descriptor and an object handle associated

with the object (also referred to hereinafter as a remote object)...." (Final Office Action,

p. 3; *Pelegri-Llopart et al.*, col. 8, ll. 57-60.) However, even if the object reference or

object handle of *Pelegri-Llopart et al.* could be interpreted as teaching the claimed

handle, no teaching has been shown of passing the remote object with the object

reference or object handle.

In the Advisory Action of June 19, 2006, the Examiner argued that the reference

discloses an embodiment in which:

> ...all interface descriptors sent between two machines have
> an identifier tag. The first time a given interface descriptor is
> sent, the tag is followed by a complete interface descriptor,
> as presented in the invention. The next time the sending
> machine wants to send that interface descriptor to the same
> receiving machine, the tag is sent instead.

(6/19/06 Advisory Action, p. 2; *Pelegri-Llopart et al.*, col. 9, ll. 55-61.) However, the

Examiner has shown no teaching in the reference that either the interface descriptor or

the tag of the reference is sent <u>with the second object</u>, i.e., an instance of a class described by the descriptor.

Because the Examiner has not shown a teaching of every element of claim 1 in *Pelegri-Llopart et al.*, the section 102 rejection of claim 1 is incorrect and Appellants therefore request its reversal. Claims 2-4 and 19-20 depend from claim 1 and therefore incorporate its recitations. Because the Examiner has not shown a teaching in the reference of every element of claim 1, he also has not shown a teaching of every element of dependent claims 2-4 and 19-20, and Appellants request the withdrawal of the section 102 rejections of claims 2-4 and 19-20.

Claims 5, 15, and 17, although varying in scope, contain similar recitations to those discussed above with respect to claim 1. Therefore, at least for the reasons discussed above with reference to claim 1, the Examiner has not shown a teaching of these elements of claims 5, 15, and 17 in *Pelegri-Llopart et al.* Therefore, the section 102 rejections of claims 5, 15, and 17, and their respective dependent claims, are also improper.

V.   Claims 7-9 are not anticipated by Pelegri-Llopart because the reference does not disclose receiving a first object from a sender with a descriptor of the class and a handle corresponding to the descriptor.

Claim 7 recites a method in a distributed system for interpreting a first object and a second object to a recipient, wherein the first object and the second object are instances of a class, comprising, among other things, receiving the first object from a sender with a descriptor of the class and a handle corresponding to the descriptor. In this step, three things are received from the sender: (1) the first object that is an

instance of a class; (2) a descriptor of the class; and (3) a handle corresponding to the descriptor. (See, e.g., Specification, p. 5, ll. 3-4; Fig. 5, step 510.)

The Examiner has not shown that *Pelegri-Llopart et al.* discloses the claimed subject matter, including at least this claim element. Instead, as the patent explains, the runtime stub generator of the first virtual machine receives an object handle (e.g., an address of a remote object) and an interface descriptor (i.e., a list of interfaces implemented by the remote object). (*Id.*, Fig. 10, Steps 803 and 804; col. 10, ll. 19-24.) The runtime stub generator then compares the interface descriptor to a list of interfaces supported by the first virtual machine, and generates a stub class that "represents in the local machine the remote object that is implemented in the second virtual machine...." (*Id.*, Fig. 10, Steps 808-818; col. 4, ll. 25-29; col. 10, ll. 34-40.)

In the Advisory Action of June 19, 2006, the Examiner argued that "Pelegri-Llopart teaches sending "data representing the object" by interpreting the interface descriptor as the data representing the object. (6/19/06 Advisory Action, p. 2, ¶ 3.) The Examiner also argued that "Pelegri-Llopart discloses...class descriptor [interface descriptor, i.e., col. 7, line 55 - col. 8, line 67]...." (*Id.*) As discussed above, this claim interpretation is unreasonable and should not be allowed to stand.

As clearly explained in the *Pelegri-Llopart et al.* reference, what is received in the reference is "an interface descriptor and an object handle associated with the object...." The remote object itself is not received. (*Pelegri-Llopart et al.*, col. 8, ll. 57-60.) Thus, the Examiner has not shown a teaching of <u>receiving the first object</u> from a sender <u>with a descriptor of the class and a handle</u> corresponding to the descriptor, as recited in claim 7.

Because the Examiner has not shown a teaching of every element of claim 7 in

Pelegri-Llopart et al., the section 102 rejection of claim 7 is improper and Appellants

request its reversal. Claims 8-9 depend from claim 7 and incorporate its recitations.

Because the Examiner has not shown a teaching in the reference of every element of

claim 7, he also has not shown a teaching of every element of dependent claims 8-9,

and Appellants request the reversal of the section 102 rejections of claims 8-9.

VI.   Claims 7-9 are not anticipated by Pelegri-Llopart because the reference does not disclose receiving the second object with the handle.

The method of claim 7 recites a combination of steps, including a step of

receiving the second object with the handle. The Examiner has not shown that *Pelegri-*

Llopart et al. teaches the claimed subject matter, including this step. Instead, the

Examiner noted that *Pelegri-Llopart et al.* discloses passing an "object reference [that]

includes an interface descriptor and an object handle associated with the object (also

referred to hereinafter as a remote object)...." (Final Office Action, p. 3; *Pelegri-Llopart*

et al., col. 8, ll. 57-60.) However, even if the object reference or object handle of

Pelegri-Llopart et al. could be interpreted as teaching the claimed handle, no teaching

has been shown of receiving <u>the remote object with</u> the object reference or object

handle.

In the Advisory Action of June 19, 2006, the Examiner argued that the reference

discloses an embodiment in which:

> ...all interface descriptors sent between two machines have
> an identifier tag. The first time a given interface descriptor is
> sent, the tag is followed by a complete interface descriptor,
> as presented in the invention. The next time the sending
> machine wants to send that interface descriptor to the same
> receiving machine, the tag is sent instead.

(6/19/06 Advisory Action, p. 2; *Pelegri-Llopart et al.*, col. 9, ll. 55-61.) However, the Examiner has shown no teaching in the reference that either the interface descriptor or the tag of the reference is received <u>with the second object</u>, i.e., an instance of a class described by the descriptor.

Because the Examiner has not shown a teaching of every element of claim 7 in *Pelegri-Llopart et al.*, the section 102 rejection of claim 7 is improper and Appellants therefore request its reversal. Claims 8-9 depend from claim 7 and incorporate its recitations. Because the Examiner has not shown a teaching in the reference of every element of claim 7, he also has not shown a teaching of every element of dependent claims 8-9, and Appellants request the withdrawal of the section 102 rejections of claims 8-9.

Conclusion

For the reasons given above, pending claims 1-9, 15, 17, and 19-22 are allowable and reversal of the Examiner's rejections is respectfully requested.

To the extent any extension of time under 37 C.F.R. § 1.136 is required to obtain entry of this Appeal Brief, such extension is hereby respectfully requested. If there are any fees due under 37 C.F.R. §§ 1.16 or 1.17 which are not enclosed herewith, including any fees required for an extension of time under 37 C.F.R. § 1.136, please charge such fees to our Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: December 12, 2006
By: Erika H. Arner
Reg. No. 57,540

25

Claims Appendix to Appeal Brief Under Rule 41.37(c)(1)(viii)

Pursuant to 37 C.F.R. § 41.37(c)(1)(viii), what follows is a clean copy of all pending claims on appeal.

1.   A method in a distributed system for passing a first object and a second object, wherein the first object and the second object are instances of a class, comprising the steps of:

passing the first object from a sender to a recipient with a descriptor of the class and a handle corresponding to the descriptor;

storing the handle and the descriptor received from the sender with the first object by the recipient;

passing the second object from the sender to the recipient with the handle; and

using the handle received by the recipient with the second object to access the descriptor received by the recipient with the first object.

2.   The method of claim 1, further comprising the step of:

assigning, by the sender, the handle to the descriptor of the class.

3.   The method of claim 1, further comprising the step of:

assigning, by the recipient, the handle to the descriptor of the class.

4.   The method of claim 1, further comprising the steps of:

using the descriptor by the recipient to interpret the first object; and

using the descriptor by the recipient to interpret the second object.

5.   A method in a distributed system for passing a first object and a second object to a recipient, wherein the first object and the second object are instances of a class, comprising the steps of:

passing, by a sender, the first object to the recipient with a descriptor of the class and a handle corresponding to the descriptor; and

passing, by the sender, the second object to the recipient with the handle, whereupon receipt by the recipient, the recipient uses the handle received with the second object to access the descriptor of the class received with the first object.

6.   The method of claim 5, further comprising the step of:

assigning the handle to the descriptor of the class.

7.   A method in a distributed system for interpreting a first object and a second object, wherein the first object and the second object are instances of a class, comprising the steps of:

receiving the first object from a sender with a descriptor of the class and a handle corresponding to the descriptor;

storing the handle and the descriptor;

receiving the second object with the handle; and

using the handle received with the second object to access the descriptor received with the first object.

8.   The method of claim 7, further comprising the step of:

assigning the handle to the descriptor of the class.

9.   The method of claim 7, further comprising the steps of:

using the descriptor to interpret the first object; and

using the descriptor to interpret the second object.

15.   A distributed system comprising:

a client computer, comprising:

a memory with a client program that sends an object of a class to a

remote location together with a handle corresponding to a descriptor of the class, and

with an outgoing serialization context that stores the descriptor of the class and the

handle corresponding to the descriptor; and

a processor that runs the client program; and

a server computer, comprising:

a memory with an incoming serialization context that stores the descriptor

of the class and the handle received from the client computer before the object was

sent, and with a server program that receives the object from the client program and

that uses the handle received with the object to access the descriptor of the class in the

incoming serialization context; and

a processor that runs the server program.

17.   A computer-readable medium containing instructions for controlling a data processing system to perform a method, the method for sending a first object and a second object from a source to a destination, wherein the first object and the second object are instances of a class, the method comprising the steps of:

sending the first object from the source to the destination with a descriptor of the class and a handle corresponding to the descriptor;

storing the handle and the descriptor received from the source by the destination;

sending the second object from the source to the destination with the handle; and

using the handle received by the destination with the second object to access the descriptor received by the destination with the first object.

19.   The method of claim 1, further comprising the step of:

creating a serialization context including the handle, the descriptor, and an indicator of whether the serialization context has been sent to the sender.

20.   The method of claim 1, further comprising the step of:

determining whether the class descriptor is accessible to the recipient.

21.   The method of claim 5, further comprising the step of:

creating a serialization context including the handle, the descriptor, and an indicator of whether the serialization context has been sent to the sender.

22.   The method of claim 5, further comprising the step of:

determining whether the class descriptor is accessible to the recipient.

Evidence Appendix to Appeal Brief Under Rule 41.37(c)(1)(ix)

No evidence is being cited in the Appeal Brief or relied upon by Appellants in the appeal.

Application No.: 09/753,686
Attorney Docket No.: 06502.0267-00

Related Proceedings Appendix to Appeal Brief Under Rule 41.37(c)(1)(x)

There are no related proceeding decisions being cited in the Appeal Brief.